FIG. 1

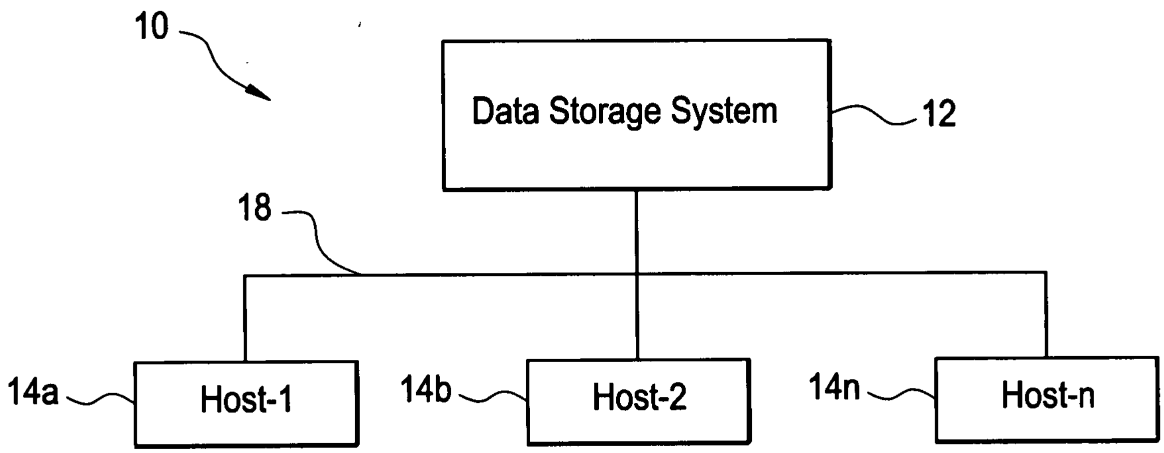

10
Data Storage System
12
18
14a
Host-1
14b
Host-2
14n
Host-n

FIG. 2

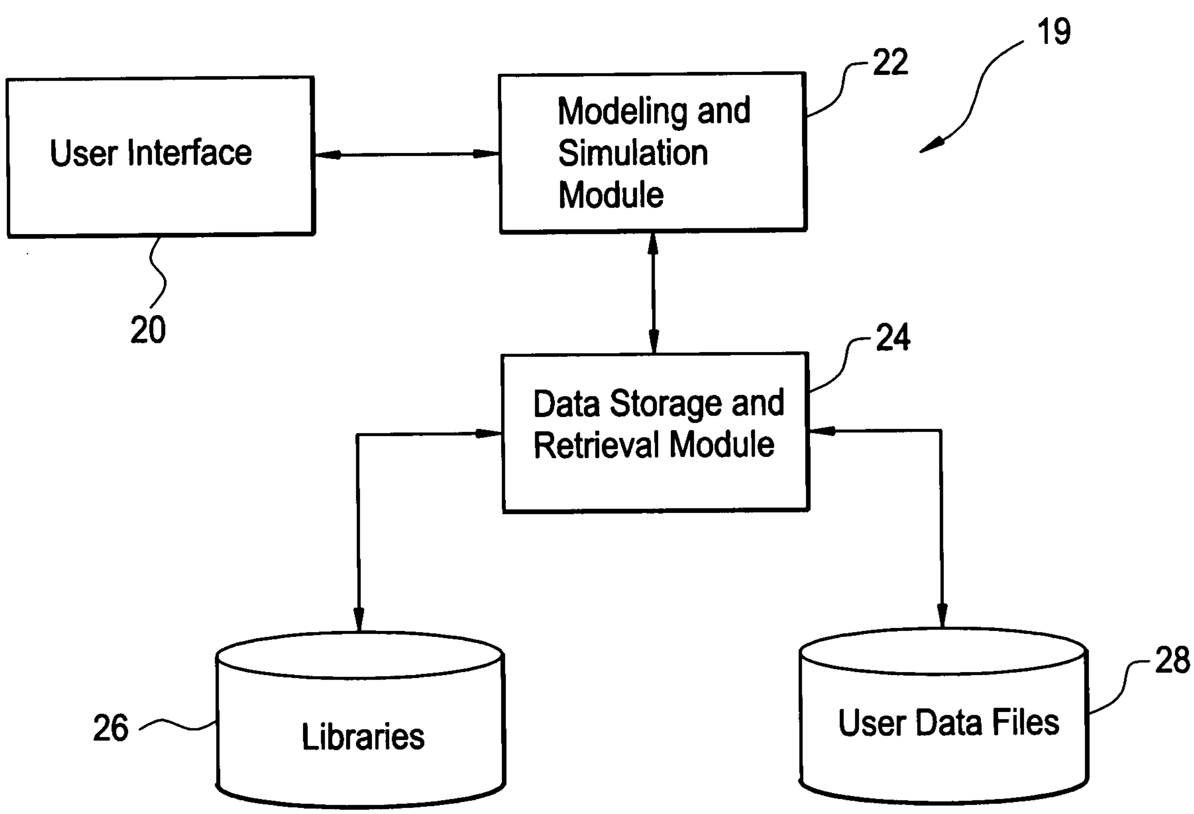

19
User Interface
20
Modeling and Simulation Module
22
Data Storage and Retrieval Module
24
Libraries
26
User Data Files
28

FIG. 3

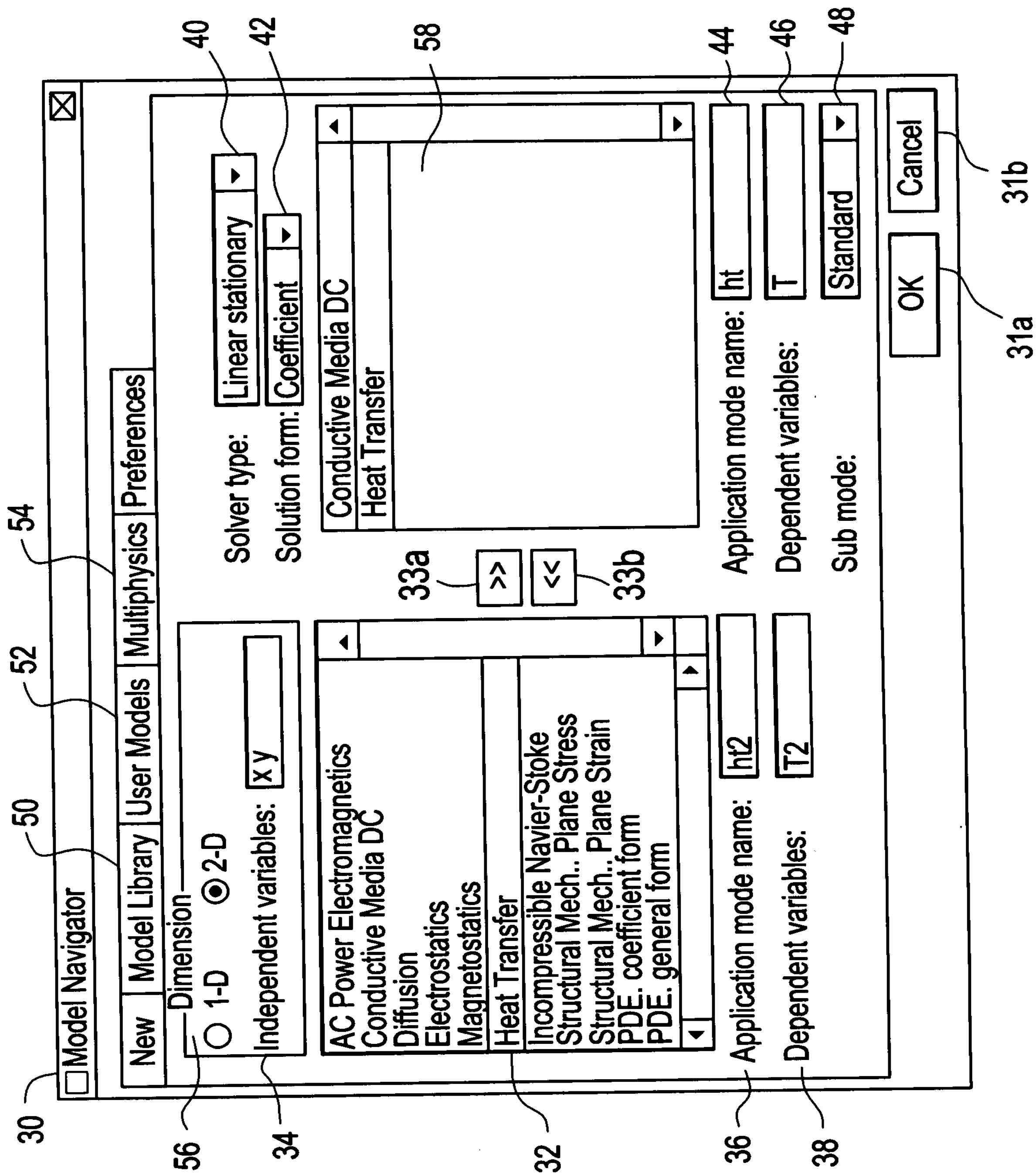

Model Navigator
New
Model Library
User Models
Multiphysics
Preferences
Dimension
1-D
2-D
Independent variables:
x y
AC Power Electromagnetics
Conductive Media DC
Diffusion
Electrostatics
Magnetostatics
Heat Transfer
Incompressible Navier-Stoke
Structural Mech.. Plane Stress
Structural Mech.. Plane Strain
PDE. coefficient form
PDE. general form
Application mode name:
ht2
Dependent variables:
T2
>>
<<
Solver type:
Linear stationary
Solution form:
Coefficient
Conductive Media DC
Heat Transfer
Application mode name:
ht
Dependent variables:
T
Sub mode:
Standard
OK
Cancel
30
50
52
54
56
34
32
36
38
33a
33b
40
42
58
44
46
48
31a
31b

FIG. 4

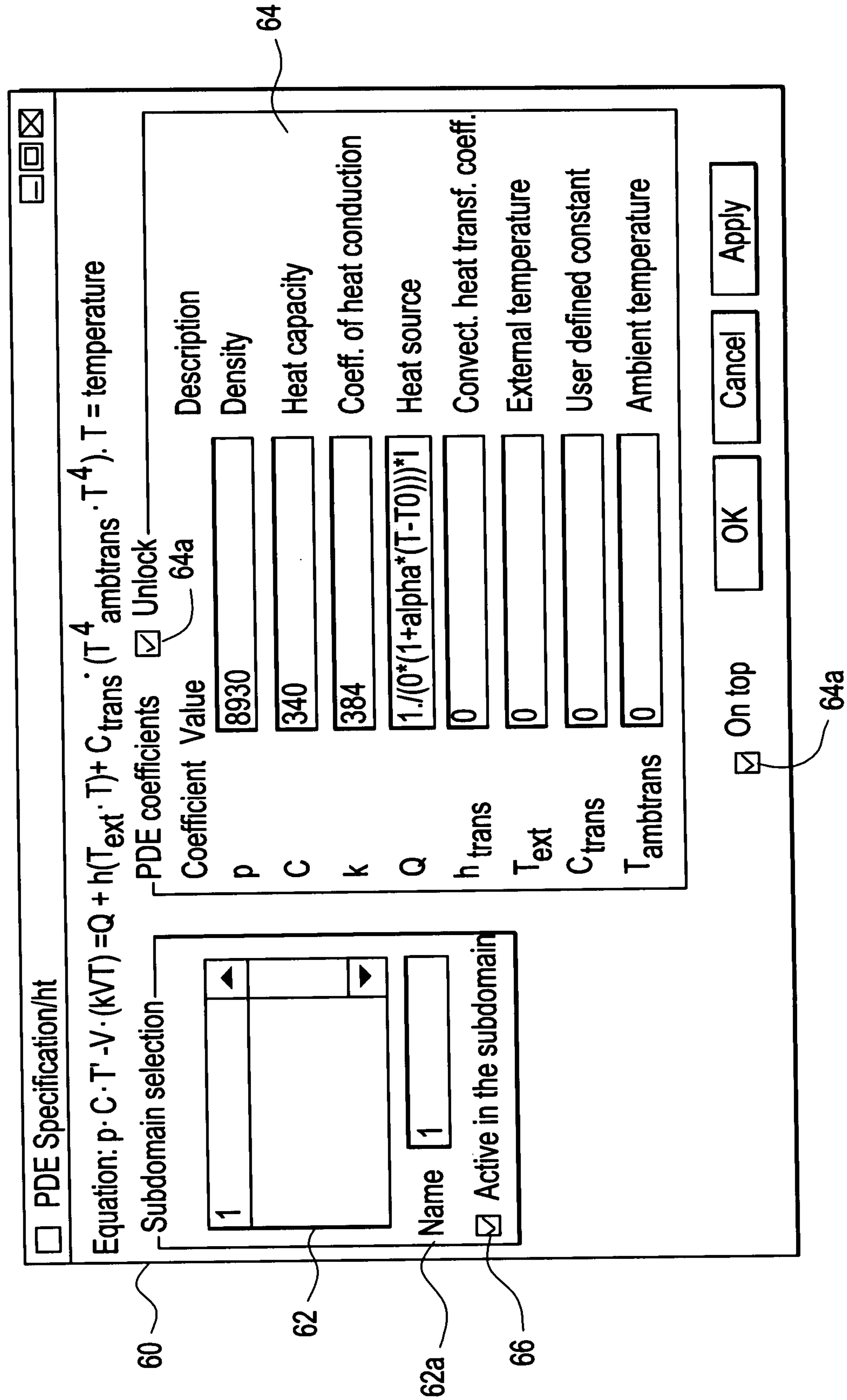

PDE Specification/ht
Equation: ρ·C·T'-∇·(k∇T) =Q + h(Text·T)+ Ctrans·(T4ambtrans·T4). T = temperature
Subdomain selection
1
Name
1
Active in the subdomain
PDE coefficients
Unlock
Coefficient
Value
Description
ρ
8930
Density
C
340
Heat capacity
k
384
Coeff. of heat conduction
Q
1./(0*(1+alpha*(T-T0)))*l
Heat source
htrans
0
Convect. heat transf. coeff.
Text
0
External temperature
Ctrans
0
User defined constant
Tambtrans
0
Ambient temperature
On top
OK
Cancel
Apply
60
62
62a
66
64
64a
64a

FIG. 5

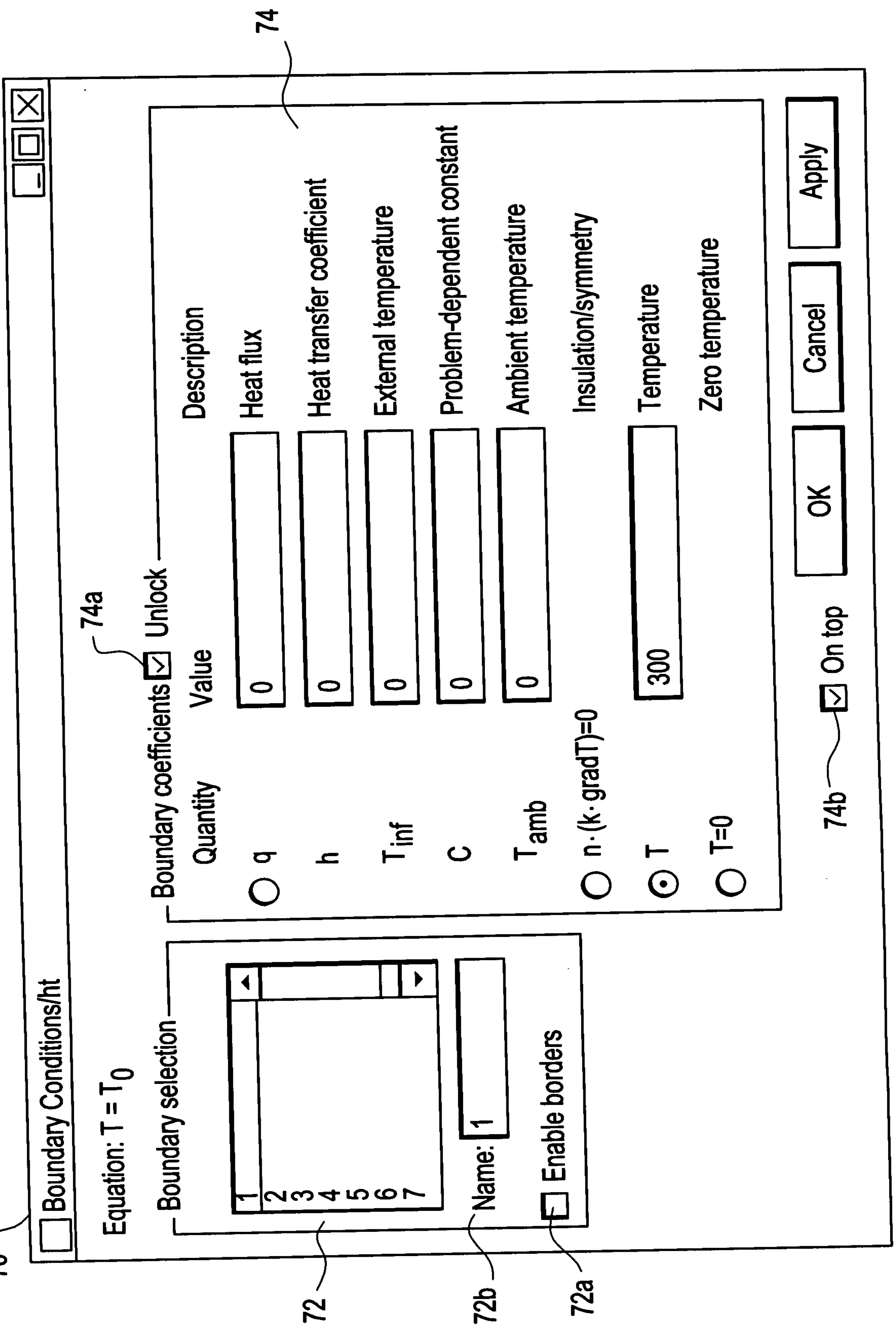

70
Boundary Conditions/ht
Equation: T = T0
Boundary selection
1
2
3
4
5
6
7
Name: 1
Enable borders
72
72b
72a
Boundary coefficients
74a
Unlock
Quantity
Value
Description
q
0
Heat flux
h
0
Heat transfer coefficient
Tinf
0
External temperature
C
0
Problem-dependent constant
Tamb
0
Ambient temperature
n·(k·gradT)=0
Insulation/symmetry
T
300
Temperature
T=0
Zero temperature
74
74b
On top
OK
Cancel
Apply

FIG. 6

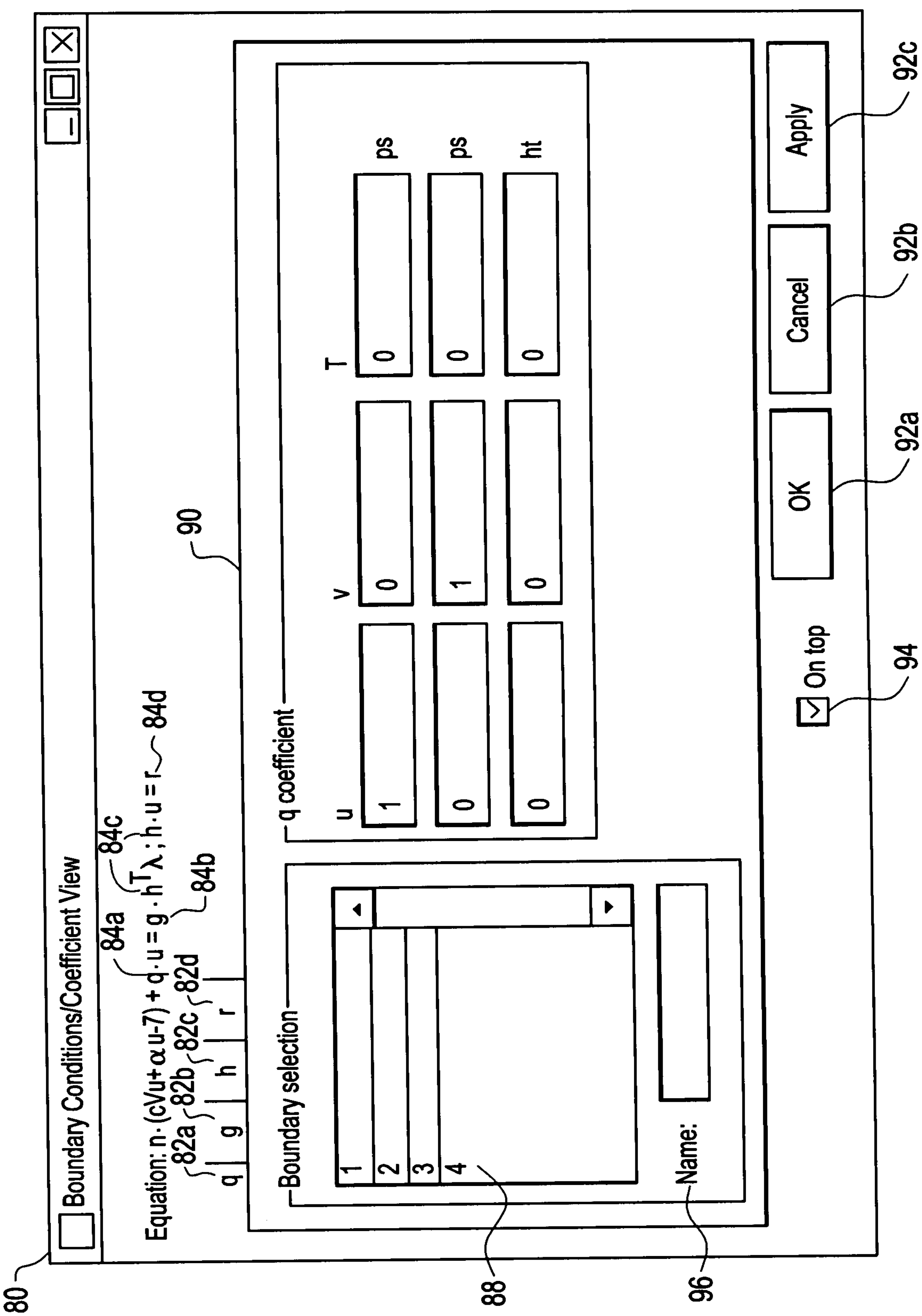

Boundary Conditions/Coefficient View
Equation: n·(cVu+αu-7)+q·u=g-h^Tλ; h·u=r
84a
84b
84c
84d
82a
82b
82c
82d
q
g
h
r
q coefficient
u
v
T
1
0
0
0
1
0
0
0
0
ps
ps
ht
Boundary selection
1
2
3
4
Name:
On top
OK
Cancel
Apply
80
88
90
92a
92b
92c
94
96

FIG. 6A

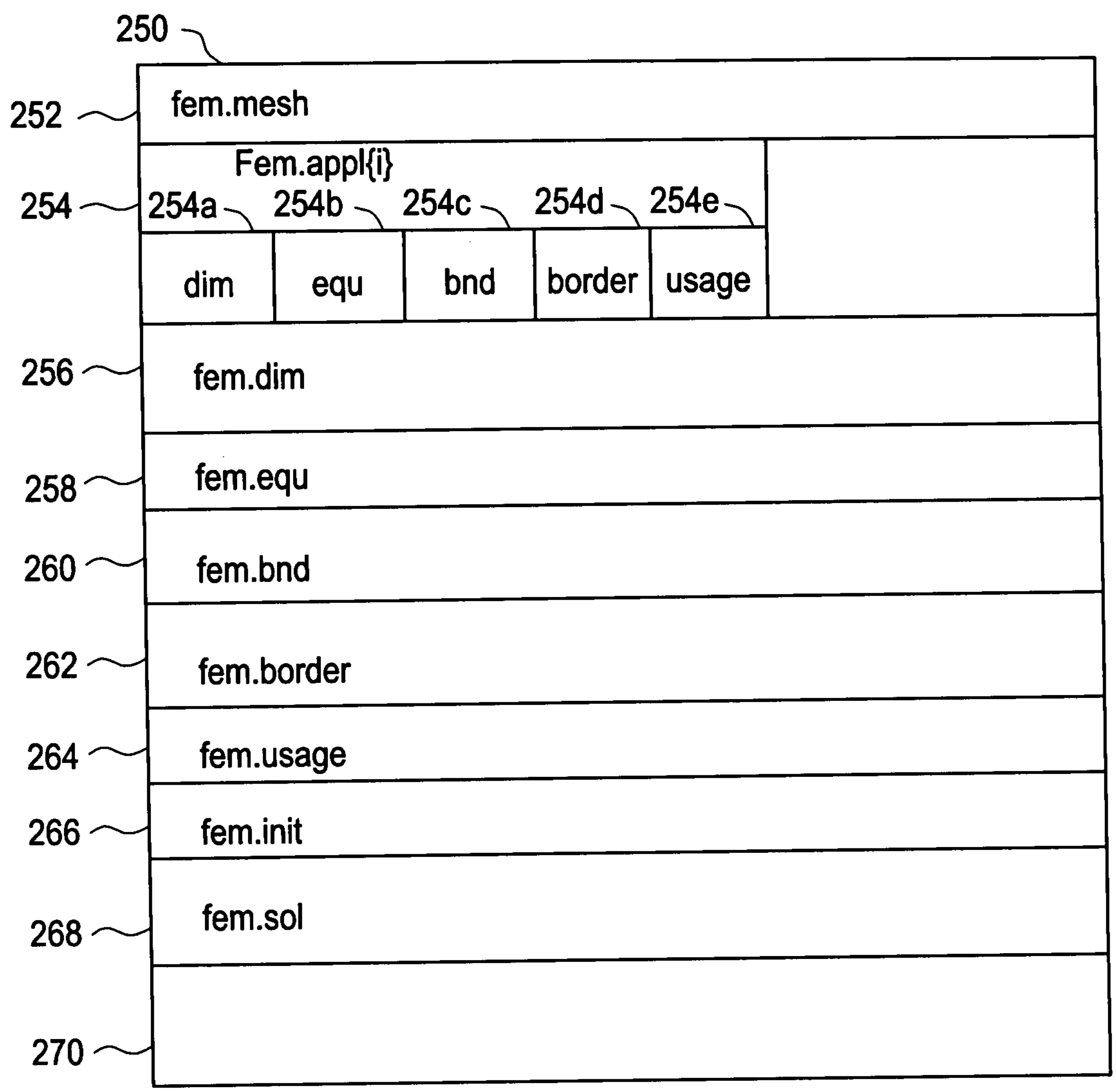
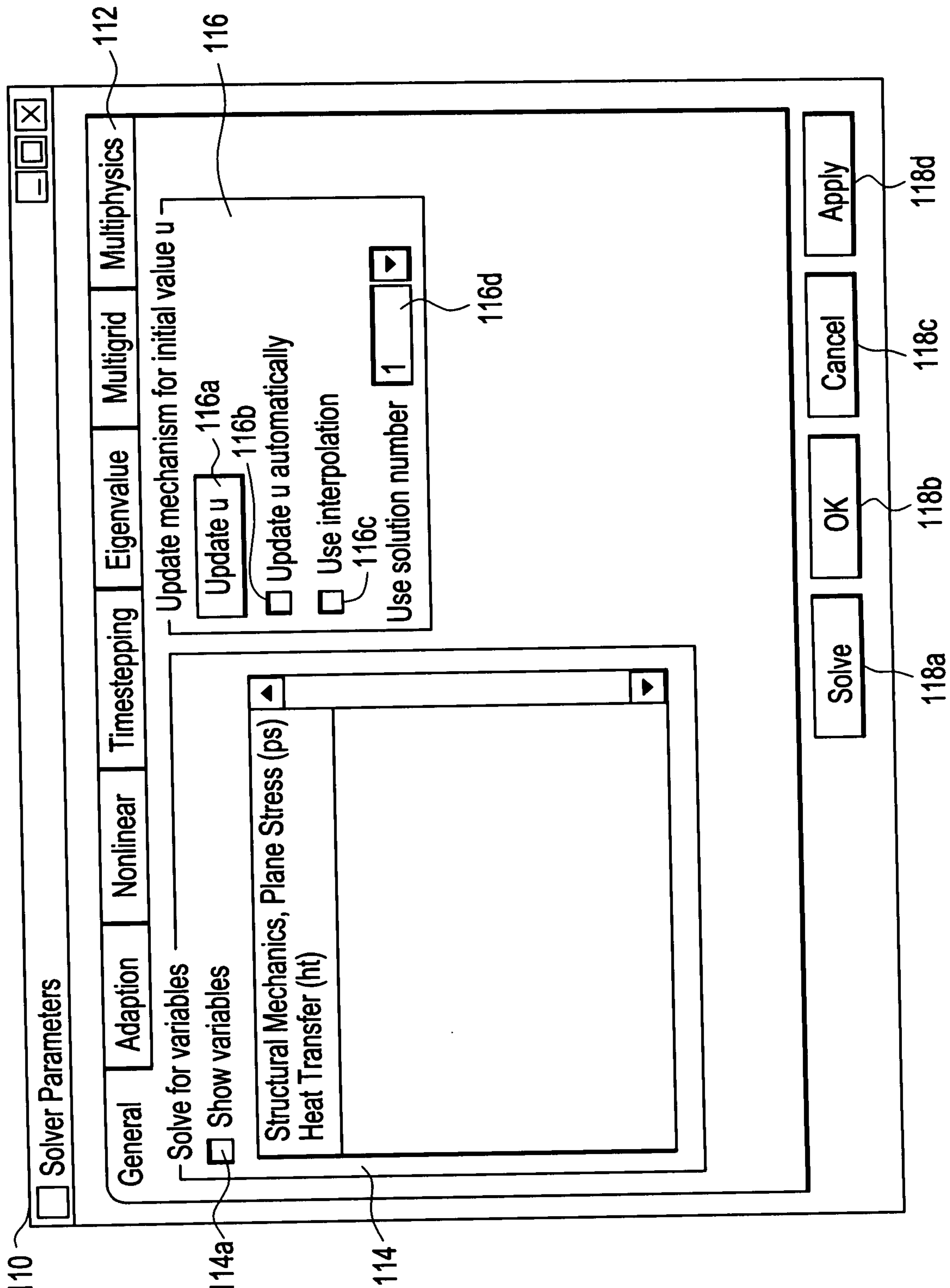

250
252
fem.mesh
254
Fem.appl{i}
254a
254b
254c
254d
254e
dim
equ
bnd
border
usage
256
fem.dim
258
fem.equ
260
fem.bnd
262
fem.border
264
fem.usage
266
fem.init
268
fem.sol
270

FIG. 7


Solver Parameters
General
Adaption
Nonlinear
Timestepping
Eigenvalue
Multigrid
Multiphysics
Solve for variables
Show variables
Structural Mechanics, Plane Stress (ps)
Heat Transfer (ht)
Update mechanism for initial value u
Update u
Update u automatically
Use interpolation
Use solution number
1
Solve
OK
Cancel
Apply
110
112
114
114a
116
116a
116b
116c
116d
118a
118b
118c
118d

Application No.: 09/675,778
Filed: June 11, 2007
Attorney Docket No.: 801939-000101
Attorney: Peter J. Prommer
Telephone No.: (312) 425-3900
Sheet 8 of 24

FIG. 8

140

$$d_{a\,lk}\frac{\partial u_k}{\partial t}-\frac{\partial}{\partial x_j}\left(c_{lkji}\frac{\partial u_k}{\partial x_i}+\alpha_{lkj}u_k-\gamma_{lj}\right)+\beta_{lki}\frac{\partial u_k}{\partial x_i}+a_{lk}u_k=f_l \quad \Omega$$

142

146a $\partial\Omega$

$$n_j\left(c_{lkji}\frac{\partial u_k}{\partial x_i}+\alpha_{lkj}u_k-\gamma_{lj}\right)+q_{lk}u_k=g_l-h_{ml}\lambda_m|$$

146b $\partial\Omega$

146

$$h_{ml}u_l=r_m$$

FIG. 9

150

$$d_{a\,lk}\frac{\partial u_k}{\partial t}+\frac{\partial \Gamma_{lj}}{\partial x_j}=F_l \quad \Omega$$

152

154a $\partial\Omega$

$$-n_j\Gamma_{lj}=G_l+\frac{\partial R_m}{\partial u_l}\lambda_m$$

154b $\partial\Omega$

154

$$0=R_m$$

FIG. 10

324

$$\begin{cases} \gamma_{lj} = \Gamma_{lj} & f_l = F_l \\ c_{lkji} = -\dfrac{\partial \Gamma_{lj}}{\partial\left(\dfrac{\partial u_k}{\partial x_i}\right)} & \alpha_{lkj} = -\dfrac{\partial \Gamma_{lj}}{\partial u_k} \\ \beta_{lki} - \dfrac{\partial F_l}{\partial\left(\dfrac{\partial u_k}{\partial x_i}\right)} & a_{lk} = -\dfrac{\partial F_l}{\partial u_k} \\ g_l = G_l & r_l = R_l \\ q_{lk} = -\dfrac{\partial G_l}{\partial u_k} & h_{lk} = -\dfrac{\partial R_l}{\partial u_k} \end{cases}$$

FIG. 11

240

$$\begin{cases} \Gamma_{lj} = c_{lkji} \dfrac{\partial u_k}{\partial x_I} \alpha_{lkj} u_k + \gamma_{lj} \\ F_I = f_I - \beta_{lki} a_{lk} u_k \\ G_l = g_l - q_{lk} u_k \\ R_m = r_m - h_{ml} u_l \end{cases}$$

FIG. 12

300

$$\begin{cases} \int_{\Omega} \left(\left(c_{lkji} \frac{\partial u_k}{\partial x_i} + \alpha_{lkj} u_k \right) \frac{\partial v}{\partial x_j} + \left(d_{a\,lk} \frac{\partial u_k}{\partial t} + \beta_{lki} \frac{\partial u_k}{\partial x_i} + a_{lk} u_k \right) v \right) dx + \\ \int_{\partial\Omega} q_{lk} u_k \, vds = \int_{\Omega} \left(Y_{lj} \frac{\partial v}{\partial x_j} + f_l v \right) dx + \int_{\partial\Omega} (g_l - h_{ml} \lambda_m) \, v\, ds \\ \int_{\partial\Omega} \mu h_{mk} u_k \, ds = \int_{\partial\Omega} \mu r_m \, ds \end{cases}$$

FIG. 13

302

$$\begin{cases} \int_{\Omega} \left(\Gamma_{lj} \frac{\partial v}{\partial x_j} + F_l v - d_{a_{lk}} \frac{\partial u_k}{\partial t} v \right) dx + \int_{\partial\Omega} \left(G_l + \frac{\partial R_m}{\partial u_l} \lambda_m \right) vds = 0 \\ \int_{\partial\Omega} R_m \mu \, ds = 0 \end{cases}$$

FIG. 14

304

$$\left\{ U_k(x) = \sum_{l=1}^{N_p} U_{l,k} \phi_l(x), \qquad \Lambda_m(x) = \sum_{K=1}^{N} \sum_{L=1}^{n} \Lambda_{K,L,m} \Psi_{K,L}(x) \right.$$

FIG. 15

306

$$\left\{\begin{array}{l}\int_{\tau}\left(c_{lkji}\,U_{I,k}\frac{\partial\phi_I}{\partial x_i}+\alpha_{lkj}U_{l\,k}\phi_l\right)\frac{\partial\phi J}{\partial x_j}dx\;+\\ \int_{\tau}\left(d_{a\,lk}\;\frac{\partial U}{\partial I}{}_{I,k}\,\phi_i+\beta_{lkj}U_{i,k}\frac{\partial\phi I}{\partial x_I}+\alpha_{lk}\;U_{i,k}\phi_i\right)\phi_J dx\;+\\ \int_{\partial_\tau}q_{lk}\;U_{l,\,k}\,\phi_I\,\phi_J\,ds=\;\int_{\tau}\left(\gamma_{I_j}\frac{\partial\phi_J}{\partial x_j}+f_I\,\phi_J\right)dx\;+\\ \int_{\partial_\tau}(g_I-h_{m\,I}\,\Lambda_{K,L,m}\,\Psi_{K,L})\,\phi_J\,ds\end{array}\right.$$

FIG. 16

308

$$\left\{\int_{\partial_\tau}h_{m\,k}\;U_{J,k}\;\phi_I\;\Psi_{K,L}\;ds=\int_{\partial_\tau}r_m\Psi_{K,L}\;ds\right.$$

FIG. 17

312

$$\left\{\begin{array}{l}\int_{\tau}\left(\Gamma_{l,j}\frac{\partial\phi_J}{\partial xj}+F_l\,\phi_J-d\,\alpha_{lk}\frac{\partial u_k}{\partial I}\,\phi_J\right)dx+\int_{\partial_t}\left(G_I+\frac{\partial R_m}{\partial u_l}\Lambda_{K,L,m}\Psi_{K,L}\right)\phi_J\,ds=0\\ \int_{\partial\tau}R_m\Psi_{K,L}\;ds=0\end{array}\right.$$

FIG. 18

310

$$
\begin{cases}
DA_{(J,l),(I,k)} = \int_\tau d_{a\,lk}\,\phi_I\,\phi_J\,dx \\
C_{(J,l),(I,k)} = \int_\tau c_{lk\,ji}\,\frac{\partial\phi_I}{\partial x_i}\ ?\ \frac{\partial\phi_J}{\partial x_j}\,dx \\
AL_{(J,l),(I,k)} = \int_\tau \alpha_{lk\,j}\,\phi_I\ ?\ \frac{\partial\phi_J}{\partial x_j}\,dx \\
BE_{(J,l),(I,k)} = \int_\tau \beta_{lk\,i}\,\frac{\partial\phi_I}{\partial x_i}\,\phi_j\,dx \\
A_{(J,l),(I,k)} = \int_\tau a_{lk}\,\phi_I\,\phi_J\,dx \\
Q_{(J,l),(I,k)} = \int_\tau q_{lk}\,\phi_I\,\phi_J\,ds \\
GA_{(J,l)} = \int_\tau \gamma_{lj}\,\frac{\partial\phi_J}{\partial x_j}\,dx \\
F_{(J,l)} = \int_\tau f_I\,\phi_J\,dx \\
G_{(J,l)} = \int_{\partial\tau} g_I\,\phi_J\,ds \\
H_{(K,L,m),(I,k)} = \int_{\partial\tau} h_{mk}\,\phi_I\,\Psi_{K,L}\,ds \\
R_{(K,L,m)} = \int_{\partial\tau} r_m\,\Psi_{K,L}\,ds
\end{cases}
$$

FIG. 19

320

$$
\begin{cases}
DA\frac{\partial U}{\partial t} + C + AL + BE + A + Q)\,U + H^T\Lambda = GA + F + G \\
H\,U = R
\end{cases}
$$

FIG. 20

322
$$\begin{cases} DA\frac{\partial U}{\partial t} + H^T\Lambda = GA + F + G \\ R = 0 \end{cases}$$

FIG. 21

326
$$\begin{cases} J(U^{(k)})\Delta U^{(k)} = p(U^{(k)}) \\ U^{(k+1)} = U^{(k)} + \lambda_k \Delta U^{(k)} \end{cases}$$

FIG. 22

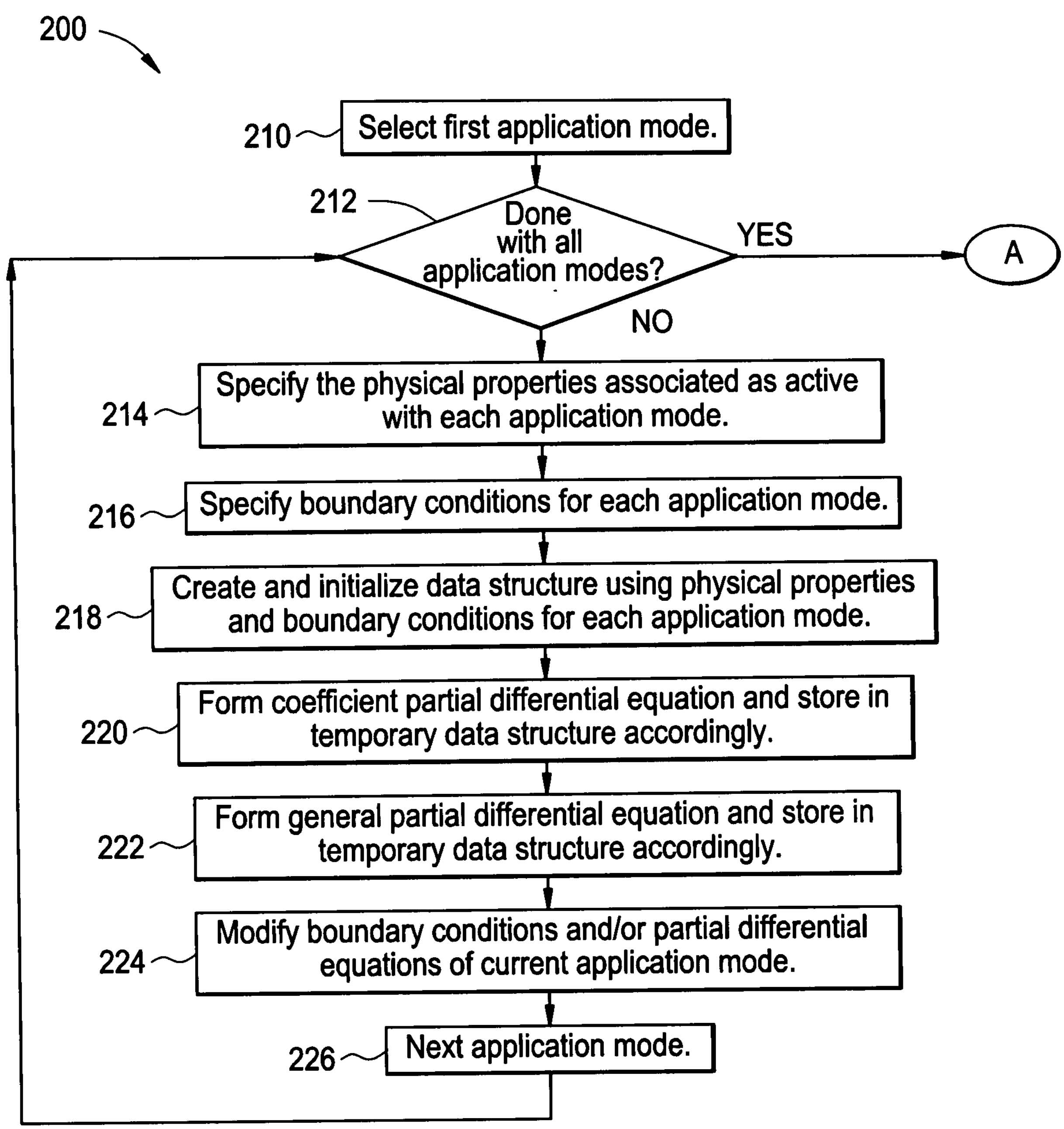

200
210
Select first application mode.
212
Done with all application modes?
YES
A
NO
214
Specify the physical properties associated as active with each application mode.
216
Specify boundary conditions for each application mode.
218
Create and initialize data structure using physical properties and boundary conditions for each application mode.
220
Form coefficient partial differential equation and store in temporary data structure accordingly.
222
Form general partial differential equation and store in temporary data structure accordingly.
224
Modify boundary conditions and/or partial differential equations of current application mode.
226
Next application mode.

FIG. 23

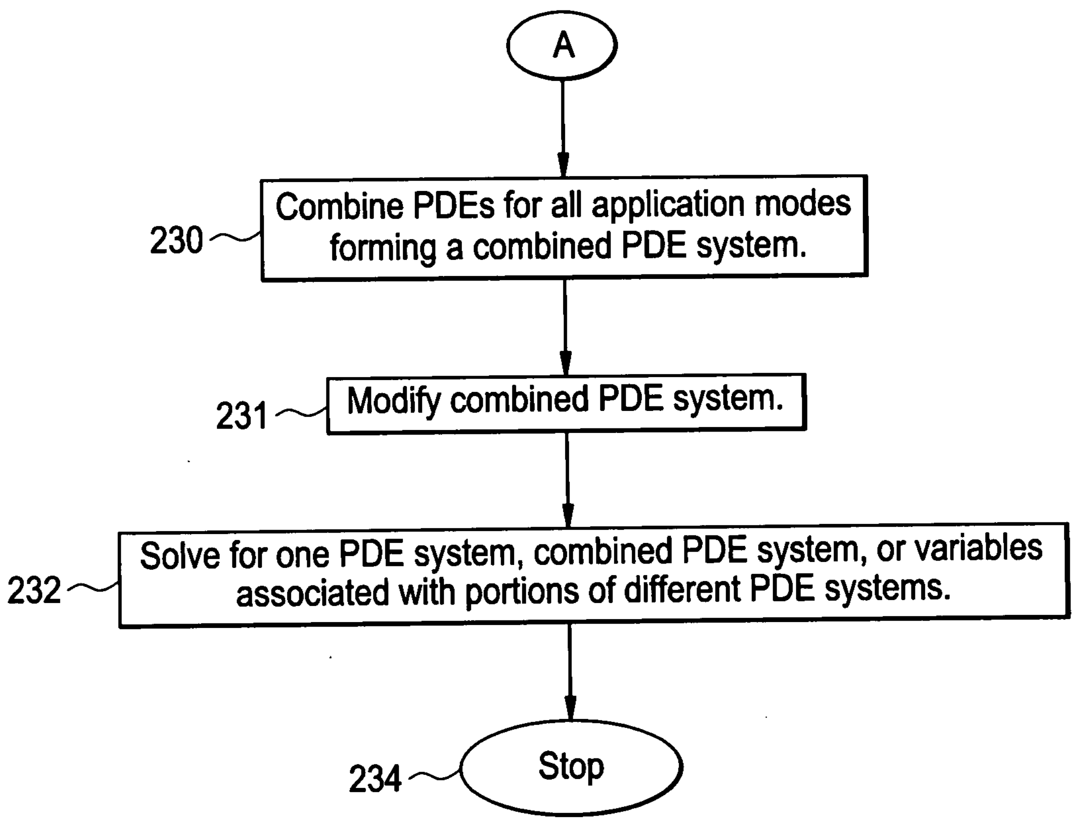

A
Combine PDEs for all application modes forming a combined PDE system.
230
Modify combined PDE system.
231
Solve for one PDE system, combined PDE system, or variables associated with portions of different PDE systems.
232
Stop
234

FIG. 24

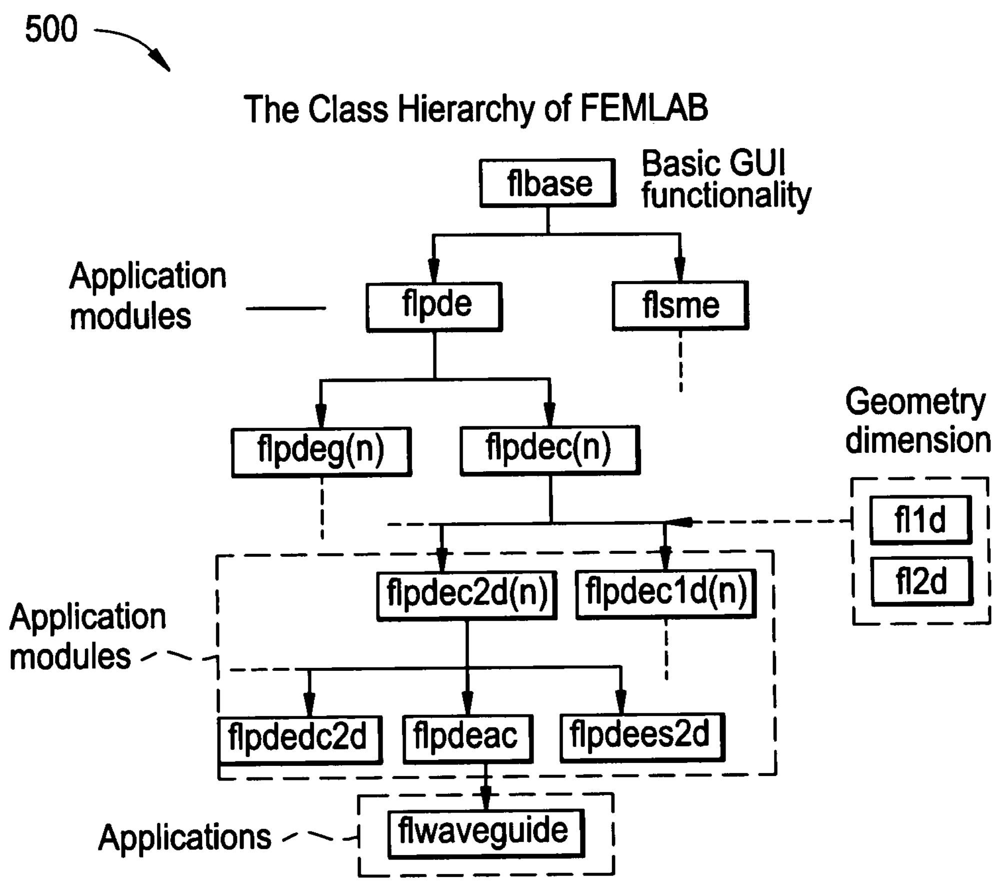

500
The Class Hierarchy of FEMLAB
flbase
Basic GUI functionality
Application modules
flpde
flsme
flpdeg(n)
flpdec(n)
Geometry dimension
fl1d
fl2d
flpdec2d(n)
flpdec1d(n)
Application modules
flpdedc2d
flpdeac
flpdees2d
Applications
flwaveguide

FIG. 25

1-D Physics Application Modes

Application mode	Class name	Parent class
Diffusion	flpdedf1d	flpdedf
Heat Transfer	flpdeht1d	flpdeht

502

1-D PDE Application Modes

Application mode	Class name	Parent class
Coefficient PDE model, n variables	flpdec1d *(n)*	flpdec *(n)*
General PDE model, n variables	flpdeg1d *(n)*	flpdeg *(n)*

504

FIG. 26

2-D Physics Application Modes

Application Mode	Class name	Parent class
AC Power Electromagnetics	flpdeac	flpdec2d
Conductive Media DC	flpdedc2d	flpdedc
Diffusion	flpdedf2d	flpdedf
Electrostatics	flpdees2d	flpdees
Magnetostatics	flpdems2d	flpdems
Heat Transfer	flpdeht2d	flpdeht
Incompressible Navier-Stokes	flpdens2d	flpdens
Structural Mechanics, Plane Stress	flpdeps	flpdec2d
Structural Mechanics, Plane Strain	flpdepn	flpdec2d

506

PDE Application Modes

Application Mode	Class name	Parent class
Coefficient PDE model, n variables	flpdec2d *(n)*	flpdec *(n)*
General PDE model, n variables	flpdeg2d *(n)*	flpdeg *(n)*

508

FIG. 27

Application Object Properties

Property name	Description	Data type
dim	Names of the dependent variables	Cell array of strings
form	PDE form	String (coefficient/ general)
name	Application name	String
parent	Parent class names	String. cell array of strings. or the empty matrix
sdim	Names of the independent variables (space dimensions)	Cell array of stings
submode	Name of current submode	String (std/wave)
tdiff	Time differentiation flag	String (on/off)

514

FIG. 28

512

```
function obj = myapp()
%MYAPP Constructor for a FEMLAB application object.

obj. name = 'My first FEMLAB application';
obj. parent = 'flpdeht2d';


% MYAPP is a subclass of FLPDEHT2D:

p1 = "flpdeht2d;
obj = class (obj, 'myapp' , p1);
set (obj , 'dim' , default_dim (obj));
```

FIG. 29

Physics Modeling Methods

514

Function	Purpose
appspec	Return application specifications.
bnd_compute	Convert application-dependent boundary conditins to generic boundary coefficents.
default_bnd	Default boundary conditions.
default_dim	Default names of dependent variables
default_equ	Default PDE coefficients/Material parameters.
default_init	Default initial conditions.
default_sdim	Default space dimension variables.
default_var	Default application scalar variables.
dim_compute	Return dependent variables for an application.
equ_compute	Convert application-dependent material parameters to generic PDE coefficients.
form_compute	Return PDE form.
init_compute	Convert application-dependent initial conditions to generic initial conditions
posttable	Define assigned variable names and post-processing information.

FIG. 30

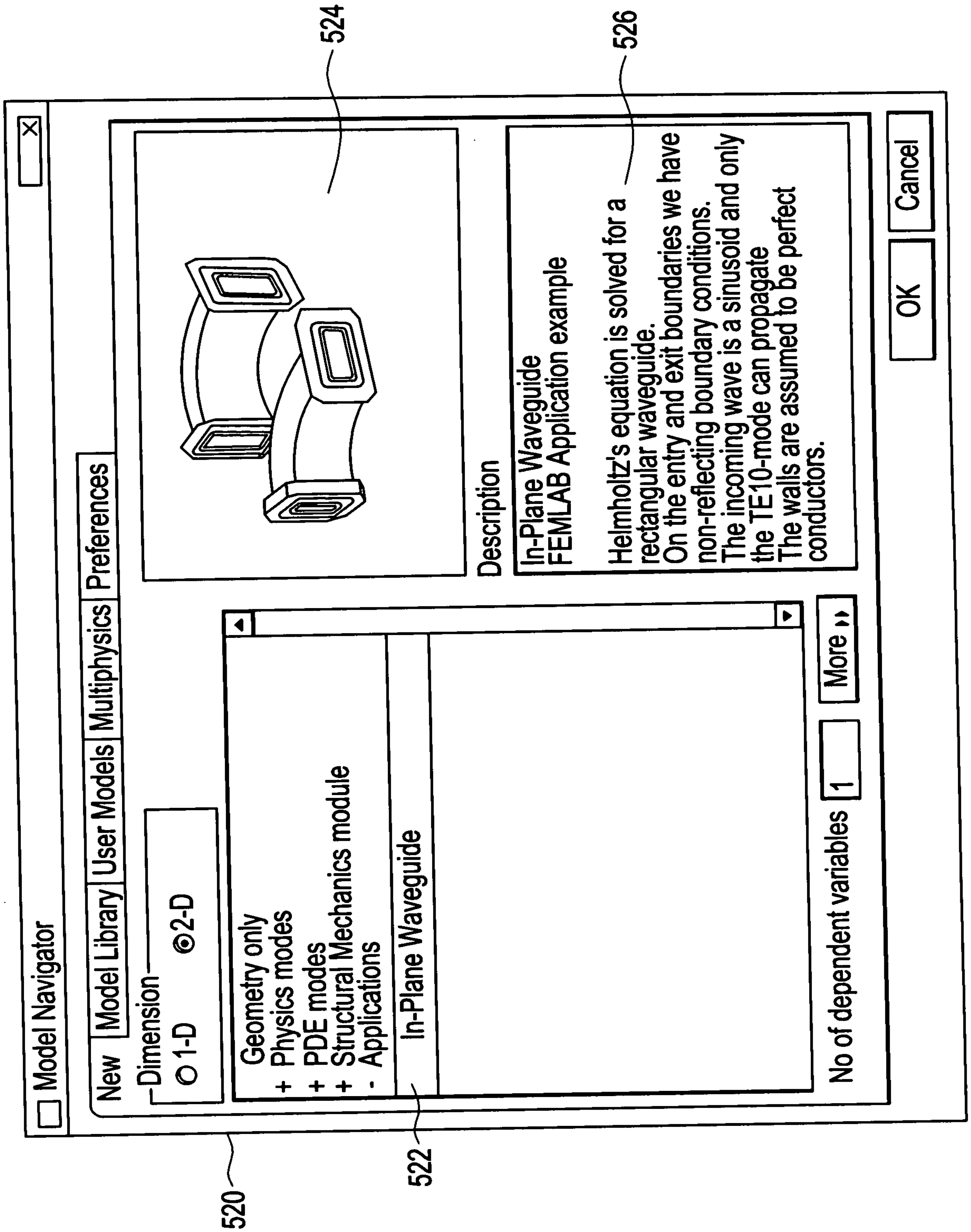

Model Navigator
New
Model Library
User Models
Multiphysics
Preferences
Dimension
1-D
2-D
Geometry only
+ Physics modes
+ PDE modes
+ Structural Mechanics module
- Applications
In-Plane Waveguide
No of dependent variables
1
More
Description
In-Plane Waveguide
FEMLAB Application example
Helmholtz's equation is solved for a rectangular waveguide.
On the entry and exit boundaries we have non-reflecting boundary conditions.
The incoming wave is a sinusoid and only the TE10-mode can propagate
The walls are assumed to be perfect conductors.
OK
Cancel
520
522
524
526

FIG. 31

530 $$\Delta E_Z + (2\pi ik)^2 E_Z = 0$$

532 $$k = \frac{1}{\lambda} = \frac{f}{c}$$

534 $$\overline{n} \cdot (\nabla E_Z) + 2\pi ik_X E_Z = 4\pi ik_X \sin\left(\frac{\pi}{d}(y - y_0)\right)$$

536 $$k^2 = k_x^2 + k_y^2$$

538 $$k_x = \sqrt{\frac{1}{\lambda^2}\ \frac{1}{(2d)^2}}$$

540 $$n \cdot (\nabla E_Z) + 2\pi ik_X E_Z = 0$$

542 $$E_Z = 0$$

544 $$f_c\ \frac{c}{2d}$$

FIG. 32

550

```
function obj = flwaveguid (varargin)
%FLWAVEGUIDE Constructor for a waveguide application object.

obj. name = 'In-Plane Waveguide';
obj. parent = 'flpdeac';

% FLWAVEGUIDE is a subclass of FLPDEAC:
p1 = flpdeac;
obj = class (obj), 'flwaveguide' ,p1);
set (obj), 'dim' , default_dim(obj));
```

FIG. 33

552

fem.user fields

Field	Description
geomparam	1-by-2 structure of geometry parameters.
entrybnd	Index to the entry boundary
exitbnd	Index to the exit boundary
freqs	Frequency vector

FIG. 34

554

fem.user fields

Field	Description
startpt	Index of the lower left corner point of the waveguide.
type	Type of waveguide. (*straight* or *elbow*)

FIG. 35

geomparam fields

556

Field	Description	Defaults for elbow	Defaults for straight
entrylength	Length of the entrance part of the waveguide.	0.1	0.1
exitlength	Length of the exit part of the waveguide.	0.1	Not used
radius	Outer radius of the waveguide bend.	0.05	Not used
width	Width of the waveguide.	0.025	0.025
cavityflag	Turn resonance cavity *on* or *off*	0	0
cavitywidth	Width of the resonance cavity	0.025	0.025
postwidth	Width of the protruding posts.	0.005	0.005
postdepth	Depth of the protruding posts.	0.005	0.005